S&H Form: PTO/SB/05 (12/97)

UTILITY PATENT APPLICATION TRANSMITTAL

(Only for new nonprovisional applications under 37 CFR 1.53(b))

Attorney Docket No.	1614.1024
First Named Inventor or Application Identifier: Yuji KUMAKURA	
Express Mail Label No.	

APPLICATION ELEMENTS
See MPEP chapter 600 concerning utility patent application contents.

ADDRESS TO: **Assistant Commissioner for Patents**
Box Patent Application
Washington, DC 20231

1. [X] Fee Transmittal Form
2. [X] Specification, Claims & Abstract ..... [Total Pages: 30]
3. [X] Drawing(s) *(35 USC 113)* ..... [Total Sheets: 9]
4. [X] Oath or Declaration ..... [Total Pages: 3]
   a. [X] Newly executed (original or copy)
   b. [ ] Copy from a prior application (37 CFR 1.63(d)) *(for continuation/divisional with Box 17 completed)*
      i. [ ] DELETION OF INVENTOR(S)
      Signed statement attached deleting inventor(s) named in the prior application, see 37 CFR 1.63(d)(2) and 1.33(b).
5. [ ] Incorporation by Reference (usable if Box 4b is checked)
   The entire disclosure of the prior application, from which a copy of the oath or declaration is supplied under Box 4b, is considered as being part of the disclosure of the accompanying application and is hereby incorporated by reference therein.
6. [ ] Microfiche Computer Program *(Appendix)*
7. [ ] Nucleotide and/or Amino Acid Sequence Submission *(if applicable, all necessary)*
   a. [ ] Computer Readable Copy
   b. [ ] Paper Copy (identical to computer copy)
   c. [ ] Statement verifying identity of above copies

ACCOMPANYING APPLICATION PARTS

8. [X] Assignment Papers (cover sheet & document(s))
9. [ ] 37 CFR 3.73(b) Statement *(when there is an assignee)* [ ] Power of Attorney
10. [ ] English Translation Document *(if applicable)*
11. [ ] Information Disclosure Statement (IDS)/PTO-1449[ ] Copies of IDS Citations
12. [ ] Preliminary Amendment
13. [X] Return Receipt Postcard (MPEP 503) *(Should be specifically itemized)*
14. [ ] Small Entity Statement(s) [ ] Statement filed in prior application, status still proper and desired.
15. [X] Certified Copy of Priority Document(s) *(if foreign priority is claimed)*
16. [ ] Other:

17. If a CONTINUING APPLICATION, *check appropriate box and supply the requisite information:*

[ ] Continuation [ ] Divisional [ ] Continuation-in-part (CIP) of prior application No: ___/_____

18. CORRESPONDENCE ADDRESS

STAAS & HALSEY LLP
Attn: H. J. Staas
700 Eleventh Street, N.W., Suite 500
Washington, DC 20001

Telephone: (202) 434-1500
Facsimile: (202) 434-1501

SPECIFICATION

TO ALL WHOM IT MAY CONCERN:

BE IT KNOWN THAT I, Yuji Kumakura, a citizen of Japan residing at Nagaoka-shi, Niigata, Japan have invented certain new and useful improvements in

INFORMATION PROCESSOR, METHOD FOR PROCESSING INFORMATION AND COMPUTER-READABLE RECORDING MEDIUM RECORDED WITH PROGRAM CODE FOR CONTROLLING A COMPUTER TO PROCESS IMFORMATION

of which the following is a specification : -

TITLE OF THE INVENTION

INFORMATION PROCESSOR, METHOD FOR PROCESSING INFORMATION AND COMPUTER-READABLE RECORDING MEDIUM RECORDED WITH PROGRAM CODE FOR CONTROLLING A COMPUTER TO PROCESS INFORMATION

BACKGROUND OF THE INVENTION

1. Field of the Invention

The present invention generally relates to information processors, methods for processing information and computer-readable recording media recorded with program code for controlling a computer to process information in which installed applications are moved to another directory or another recording medium, and more particularly to an information processor, a method for processing information and a computer-readable recording medium recorded with program code for controlling a computer to process information in which it is possible to copy an installed application including related data from a current directory to another directory or another recording medium and continuously to delete the application including related data in the current directory.

Recently, in the personal computer (hereinafter called a PC) industry, a storage device has been developed and a capacity of the storage device is becoming much larger so that a large-sized operating system and various large-sized applications can be installed in the storage device. With increasing capacities of applications, data belonging to the applications are becoming diversified and also large sized. In this state, actually, the existing storage device does not have enough capacity to manage the applications and data belonging thereto. Usually, many PC users install an additional internal or external storage device to

their PCs. In addition, the PC users reinstall the same applications installed in the existing storage device to the new storage device.

In the above state, as a result of moving existing applications to another directory, it is required that the installed applications be uninstalled, the applications be reinstalled, and an OS (Operation System) be restarted a few times. However, these processes are not performed effectively. Especially for inexperienced users, it is not easy to complete these processes without any problems. Therefore, it is desired that a method for installing an application should be simplified.

2. Description of the Related Art

Conventionally, the following processes are required to reinstall an application in another directory or storage device.

First, a conventional installation will be explained with reference to FIG.1.

FIG.1 shows an example of a setup window for a conventional installation.

Generally, when a user installs an application, the user chooses a type of installation from a setup window and clicks a button corresponding to the type of installation so as to start to install the application.

In FIG.1, the user chooses a desired operation from a setup window 200. For instance, when the user installs an application for the first time, the user selects an "INSTALL" button 201 for the initial installation. When the user reinstalls the application including additional functions in the directory in which the application and the additional functions are already installed, the user also selects the "INSTALL" button 201.

When the user adds more functions to currently installed application and functions, the

user selects an "ADD FUNCTIONS" button 202.

When the user removes the currently installed functions, the user selects a "REMOVE FUNCTIONS" button 203.

When the user cancels the setup, the user selects a "CANCEL" button 204.

Any operation selected by the above buttons, except for the initial installation, is performed under the directory in the drive indicated at the initial installation.

When the user moves an installed application to another storage area, that is, when the user uninstalls the application and then installs the application to another storage area, the following steps are required:

step 1 : if necessary, store all data created by the application.

step 2 : execute an uninstallation program to delete the application from the storage area.

step 3 : restart an OS.

step 4 : open the setup window to click "INSTALL" button 201 in FIG.1 and indicate a destination to install the application after restarting the OS.

step 5 : restart the OS again after the installation is completed.

step 6 : if necessary, restore all data stored in the step 1 to a destination directory.

The user follows the above-mentioned steps to complete the move of the application and the data.

However, the above-mentioned conventional manner to move an application has the following disadvantages:

First, as mentioned above, the many steps to move an application take much time.

Second, a user generally changes optional

settings of the application to fit the user's requirements after the first installation. For example, font size, spacing between lines, lines per page, and the like may be the optional settings. Thus, the user has to set the optional settings again after the application is moved to another directory. In addition, the user has to restore data that is temporarily stored in another storage area. Thus, it may not be possible for the user to use the application soon after the reinstallation thereof. Actually, an inexperienced user tends to lose important data during the above steps. Therefore, it is preferable to perform the steps by an experienced user.

Third, regardless of the above steps, when a user moves an application, the user is required to properly change information for executing the application, which information is recorded in a file referred to by other applications or the OS. Hence, when the user does not properly change the information in the file, not only the application but also other applications and the OS do not perform properly.

SUMMARY OF THE INVENTION

It is a general object of the present invention to provide an information processor, a method for processing information and a computer-readable recording medium recorded with program code for controlling a computer to process information in which the above-mentioned problems are eliminated.

A more specific object of the present invention is to provide an information processor, a method for processing information and a computer-readable recording medium recorded with program code for controlling a computer to process information in which it is possible to move an application easily

and safely.

The above objects of the present invention are achieved by an information processor including: a control information retrieving part for retrieving control information that is used to execute a program; a destination defining part for defining destination address information to move the program; a moving part for moving the program in accordance with the destination address information; and a control information changing part for changing the control information based on the destination address information.

According to the present invention, it is possible to move a program as it is, without any changes of optional settings, in accordance with the destination address information. In addition, the present invention does not require a user to change the control information and also the program performs properly so that the user can use the program soon after the program is moved.

The above objects of the present invention are achieved by a method for processing information including the steps of: (a) retrieving control information that is used to execute a program; (b) defining destination address information; (c) moving the program in accordance with the destination address information; and (d) changing the control information based on the destination address information.

According to the present invention, a method is provided to move a program as it is, without any changes of optional settings, in accordance with the destination address information. By applying the method, a user does not have to change the control information and also the program performs properly so that the user can use the program soon after the program is moved.

The above objects of the present invention are achieved by a computer-readable recording medium recorded with program code for controlling a computer to process information, the program code including the codes for: (a) retrieving control information that is used to execute a program; (b) defining destination address information; (c) moving the program in accordance with the destination address information; and (d) changing the control information based on the destination address information.

According to the present invention, a computer-readable recording medium recorded with program code for controlling a computer to process information is provided to move a program as it is, without any changes of options, in accordance with the destination address information. Therefore, a user does not have to change the control information and also the program performs properly so that the user can use the program soon after the program is moved.

BRIEF DESCRIPTION OF THE DRAWINGS

Other objects, features and advantages of the present invention will become more apparent from the following detailed description when read in conjunction with the accompanying drawings, in which:

FIG.1 shows an example of a setup window for a conventional installation;

FIG.2 shows a diagram illustrating an example of a constitution of a whole system according to an embodiment of the present invention;

FIG.3 shows a diagram illustrating a hardware construction according to the embodiment of the present invention;

FIG.4A shows a diagram illustrating

registry information before an application is moved and FIG.4B shows a diagram illustrating the registry information after the application is moved;

FIG.5A shows a diagram illustrating a structure of a definition file according to the embodiment of the present invention and FIG.5B shows a diagram illustrating an example of the definition file according to the embodiment of the present invention;

FIG.6 shows a diagram illustrating a setup window according to the embodiment of the present invention;

FIG.7 shows a diagram illustrating a destination entry window according to the embodiment of the present invention;

FIG.8 shows a diagram illustrating a dialog box showing a status of the moving application process; and

FIG.9 shows a flowchart diagram illustrating the moving application process according to the embodiment of the present invention.

DESCRIPTION OF THE PREFERRED EMBODIMENTS

FIG. 2 shows a diagram illustrating an example of a constitution of a whole system according to an embodiment of the present invention.

An install apparatus 100 according to the embodiment corresponds to the information processor and includes the following parts: an installer 1 that is developed in a virtual storage to install an application; a definition file 2 to store destination information; a setup application 3 to setup; an OS 4; a display processing part 5 that controls display information; an input processing part 6 that controls data inputted by a user; an output processing part 7; a recording medium 8 that is internally mounted as a standard drive C; a

recording medium 9 such a CD-ROM in which an application is recorded and is sold as a product; and a recording medium 10 which is an external or an internal hard disk such as a drive D that is used as a destination drive when the application is moved. In addition, the recording medium 8 further includes a registry information 8a to maintain all information of installed applications, and an application 8b that performs on the OS 4. The recording medium 9 further includes setup application 9a, a definition file 9b that is developed in the virtual storage to maintain destination information during an installation, an installer 9c that is developed in the virtual storage to control the installation, and an application 9d to be installed into a PC.

It should be noted that the application 8b stored in the recording medium 8 is all or a part of the application 9d recorded in the recording medium 9.

For instance, the install apparatus 100 is required to execute the setup application 9a each time the application 8b installed in the recording medium 8 is moved to the recording medium 10. After the setup application 9a is executed, the setup application 9a, the definition file 9b, and installer 9c are developed in a virtual storage area controlled by the OS 4. In FIG.2, the installer 1, the definition file 2, and the setup application 3 are the installer 9c, the definition file 9b, and the setup application 9a developed in the virtual storage area, respectively.

The setup application 3 retrieves a current storage address of the installed application from the registry information 8a in the recording medium 8. Subsequently, the setup application 3 retrieves the destination address from the

definition file 2 and then executes the installer 1 to move the application 8b in the recording medium 8 to the destination.

The destination information inputted by the user is temporarily maintained in the definition file 2.

When the setup application 3 is terminated, the setup application 3, the installer 1, and definition file 2 are removed from the virtual storage area.

FIG.3 shows a diagram illustrating a hardware construction according to the embodiment of the present invention.

Referring to FIG.3, the install apparatus 100 is constructed with the following: a CPU 11 that executes an installation program according to the present invention (explained later); a memory unit 12 that temporarily stores instructions to execute the program and data; a storage device 13 such as a hard disk assigned to a drive C into which the program is loaded and the needed data is stored; an input unit 14 that controls data inputted by a user, a display unit 15 that controls the display information; a storage device 16 such a hard disk assigned to a drive D; and a CD-ROM 17 where the program is recorded. Of course, the medium recording the installation program according to the present invention is not limited to the CD-ROM, but other computer-readable recording media may be used.

The registry information will now be explained.

In this embodiment, the word 'folder' is used synonymously with 'directory'.

FIG.4A shows a diagram illustrating the registry information before the application is moved.

It is assumed that a name of the application 8b is "Omakase V3" and the recording

medium 8 is assigned to the drive C.

In this example, the registry information 8a includes three keys: InstallDir, DataPath, and ProgramFolder. The key "InstallDir" indicates an area storing the application 8b. That is, a directory "C:¥ProgramFiles¥OmakaseV3", in which the application "Omakase V3" is installed, corresponds to the key "InstallDir".

The key "DataPath" indicates a directory "C:¥ProgramFiles¥OmakaseV3¥Data" that stores data needed to perform the application 8b.

The key "ProgramFolder" indicates an area storing a group folder "OmakaseV3" that helps a user to execute the application "Omakase V3" without knowing where the "Omakase V3" is located in the recording medium 8.

FIG.4B shows a diagram illustrating the registry information after the application is moved.

After the application 8b is moved from the recording medium 8 to the recording medium 10, the registry information 8a is changed by the installer 1.

It is assumed that the recording medium 10 is assigned to the drive D.

When the application 8b is moved to the directory "OmakaseV3" in the drive D, the installer 1 changes information such that a directory "D:¥OmakaseV3" corresponds to the key "InstallDir", a directory "D:¥OmakaseV3¥Data" corresponds to the key "DataPath", and a group folder "OmakaseV3" corresponds to the key "ProgramFolder".

The definition file used when the registry information 8a is changed will now be explained.

FIG.5A shows a diagram illustrating a structure of the definition file according to the embodiment of the present invention.

In FIG.5A, each path information

corresponds to each directory information in the registry information 8a as shown in FIG.4. That is, path 0 and path 1 in FIG.5A correspond to the keys "InstallDir" and "DataPath" in FIG.4, respectively.

Each path information is composed of a root key, a subkey, a value name and an additional path.

The root key and the subkey indicate an address storing the registry information 8a.

The value name indicates the key name defined in the registry information 8a. Thus, the value name indicates information identically in the registry information 8a.

If necessary, the additional path is set with an additional character string composed of the root key and the subkey.

A folder information is composed of the following: an icon name, a command line, a folder, icon file, and an icon index. The folder information is used to change icon data, which is defined for a program folder or for a short cut, to indicate a destination path.

The command line is information to change a work folder or link destination information for the icon data.

The folder is information to change address information.

The icon file is information to change destination information of a program execution file.

The icon index is an icon number to indicate one of a plurality of icons. The icon number starts from 0 indicating the first icon.

FIG.5B shows a diagram illustrating an example of the definition file according to the embodiment of the present invention.

The path 0 indicates an address storing the registry information 8a by the root key and the

subkey. That is,

"MACHIN-A¥Software¥Fujitsu¥Omakase¥V3.0¥Dir"

indicates the address of the registry information 8a. The value name "InstallDir" corresponds to the key "InstallDir" in FIG.4B, and indicates an install path as an address of the installed application. The install path is defined by the installer 1.

In the same manner as for the path 0, the root key and the subkey in the path 1 indicate the address of the registry information 8a. The value name "DataPath" corresponds to the key "DataPath" in FIG.4B. Subsequently, the additional path "Data" is added to the install path so that a data path is defined.

In the folder information in the definition file 2, a variable "Install" is used to set a drive name or a directory name indicated by the user to move the application. In the definition file 2, the variable "Install" is defined between two % characters so as to be recognized as a variable.

In this embodiment, the icon name is defined as "Omakase V3" which is the application name.

The command line is defined by the variable "Install" and an execution file name "Omakase.exe".

The folder is defined by the variable "Install" which indicates the destination to which to move the application.

The icon file name is defined by the variable "Install" and the execution file name "Omakase.exe".

The icon index indicates an icon 0. For example, when there are n icons, an icon number starts from 0 and ends at n-1.

It should be noted that contents, a

structure and so on of the registry information in the embodiment are well known as that of MS-Windows™ of Microsoft Corporation. This registry information includes many kinds of information to operate hardware and software and is stored as a database file. For example, the registry information of MS-Windows™ is managed by two files: USER.DAT and SYSTEM.DAT.

FIG.6 shows a diagram illustrating a setup window according to the embodiment of the present invention. In this figure, parts that are the same as those shown in the previously described figures are given the same reference numbers.

Referring to FIG.6, in the setup window 200 according to the embodiment, a "MOVE APPLICATION" button 101 is added.

The "MOVE APPLICATION" button 101 is clicked to move the installed application with additional functions as it is to a drive or a directory indicated by a user.

This function of the "MOVE APPLICATION" does not require the user to store and restore data, and also does not require the user to install additional functions, either.

When the user clicks the "MOVE APPLICATION" button 101 in the setup window 200, a window to indicate a destination is displayed.

FIG.7 shows a diagram illustrating a destination entry window according to the embodiment of the present invention.

Referring to FIG.7, a destination entry window 20 includes a destination input area 21 to input a destination path by a user, a "REFER TO" button 22 to refer to current drives or directories, an "OK" button 23 to move the application, and a "CANCEL" button 24 to cancel the move of the application.

For example, when the user desires to move an application to the drive D to which the recording medium 10 is assigned, the user indicates the drive D in the destination input area 21. When the user does not indicate any directory, a directory "OmakaseV3", which is the same as the current directory, is created automatically in the drive D.

To indicate a destination, the user inputs a destination path directly in the destination input area 21, or the user clicks the "REFER TO" button 22 and selects from a window showing a list of current drives and directories that appear by clicking the "REFER TO" button 22.

When the user clicks the "OK" button 23 after the user decides and inputs the destination path in the destination input area 21, the installer 1 is executed and the installer 1 proceeds to a moving application process, which will be explained later.

However, when the user terminates the setup process, the user clicks the "CANCEL" button 24.

FIG.8 shows a diagram illustrating a dialog box showing a status of the moving application process.

Referring to FIG.8, a dialog box 30 includes a barometer 31 that shows a progress of the moving application process visually and a % display 32 that shows a percentage of accomplishment.

When the user clicks the "OK" button 23, the moving application process starts and the dialog box 30 is displayed simultaneously.

In the barometer 31, a dark color part extends to the right side in correspondence with the progress of the moving application process. When the dark color part reaches the end of the right side, it means that the moving application process

is completed.

The % display 32 synchronizes with the barometer 31 and shows the progress of the moving application process by a percentage.

The user can visually realize a status of the moving application process.

The moving application process will now be explained. The moving application process is performed by the install program according to the present invention.

FIG.9 shows a flowchart diagram illustrating the moving application process according to the embodiment of the present invention.

Referring to FIG.9, the moving application process includes the following steps: a step S1 to choose the "MOVE APPLICATION" button 110; a step S2 to indicate a destination folder; a step S3 to check a space capacity; a step S4 to check possibility to move; a step S5 to generate folders to copy files; a step S6 to check success in copying; a step S7 to change the registry information; a step S8 to check success in changing; a step S9 to delete original files; a step S10 to delete copied files; a step S11 to display a "COMPLETED" message or "FAILED" message; and a step S12 to restart the OS.

In the step S1, a user clicks the "MOVE APPLICATION" button 110 in the setup window 200 as shown in FIG.6, and then the moving application process starts.

In the step S2, the destination entry window 20 is displayed so that the user inputs a destination path.

When the user clicks the "OK" button 23 in FIG.7, a space capacity of the indicated destination disk is checked in the step S3.

In the step S4, when the space capacity is enough to move the application, the step S5 is

performed. However, when the space capacity is not enough to move the application, the step S11 is performed in order to display the "FAILED" message to the user.

When the space capacity is recognized as enough, new folders are created in the destination directory in the step S5. Then, the moving application process starts to copy existing files in current folders to the created folders.

The installer 1 refers to the root key and the subkey in the definition file 2 and retrieves the current install path and the current data path from the registry information 8a. That is, the installer 1 retrieves the current install path in accordance with the key "InstallDir" from the registry information 8a and then copies the application from the current install path to the destination. Subsequently, the installer 1 also retrieves the current data path in accordance with the key "DataPath" from the registry information 8a and then copies the data related to the application from the current data path to the destination.

In the step S6, it is checked as to whether copying in the step S5 succeeds or not. When the result is successful, the step S7 is performed to change current paths to new paths in the registry information 8a. When the result is negative, the step S11 is performed to terminate the process.

When copying succeeds in the step S5, the installer 1 sets the destination information indicated by the user to the variable "Install" of the folder information in the definition file 2 and then changes information in the registry information 8a. For example, it is assumed that the user indicates the recording medium 10 assigned to the drive D as the destination to move the application.

That is, the user inputs "D:¥OmakaseV3" in the destination input area 21 as shown in FIG.7. Then, the variable "Install" is replaced with "D:¥OmakaseV3". The install path indicated by the key "InstallDir" in the registry information 8a is replaced with "D:¥OmakaseV3" and the data path indicated by the key "DataPath" in the registry information 8a is replaced with "D:¥OmakaseV3¥Data" which is a character string composed of the install path and the additional path "Data".

After the path information in the registry information 8a is changed, it is checked as to whether the replacement with new information in the registry information 8a is completed successfully in the step S8. When the result of the checking is successful, the step S9 is performed to delete the application stored in the current directory. When the result is negative, the step S11 is performed to display the "FAILED" message.

After completing the change of the path information, the original files, including the application 8b in the current directory in the recording medium 8 assigned to the drive C, are deleted in the step S9.

When in replacing with new path information in the registry information 8a fails, the created files, including the copied application 8b in the recording medium 10 assigned to the drive D, are deleted in the step S10.

In the step S11, in accordance with a status of decision steps S4, S6, and S8, a "COMPLETED" message or a "FAILED" message is displayed to notify the user of the result of the process.

After displaying a message, the process restarts the OS 4 and then the process is terminated in the step S12.

As mentioned above, all of the installed applications, the installed additional functions, and created data are copied to the destination indicated by the user as it is so that the user does not need to reinstall the same additional functions and does not set again optional settings of the application to fit the user's requirements such as a font size, lines per page, and the like. In addition, a use is only required to indicate a destination so that the user does not have to uninstall the application from the current directory, install the application in a new directory, and restart the OS several times. Therefore, the present invention can reduce time consumption and perform effectively to move an application.

In this embodiment, in a case in which two recording media are assigned to two logical drives C and D, respectively, the manner of moving an application (a program) is explained. Alternately, the present invention can be applied to a case in which one recording medium is segmented into two areas and is assigned to two logical drives C and D, respectively, so that an application is moved within one and the same recording medium.

Further, the present invention can be applied to another case in which a destination directory is defined as a current directory in the same logical drive, so that an application is moved within the same logical drive. In the embodiment, an application is physically moved to another directory in another recording medium. On the contrary, in the case in which an application is moved within the same logical drive, instead of copying, it is possible to move a current application by changing management information such directory information and file information, which are managed by a file management system of the OS.

It should be noted that the present invention is related to not only a manner of moving an application (a program) physically to another recording medium but also to a manner of changing the management information of the file management system without copying the application.

In addition, when an application is installed again, information showing the fact that the application is installed is saved in an install information part that is referred to when the application is deleted by the OS. It should be noted that this technology is well known as that of MS-Windows™.

The present invention is not limited to the specifically disclosed embodiments, variations and modifications, and other variations and modifications may be made without departing from the scope of the present invention.

The present application is based on Japanese Priority Application NO. 11-054179 filed on March 2, 1999, the entire contents of which are hereby incorporated by reference.

WHAT IS CLAIMED IS:

1. An information processor comprising:

a control information retrieving part retrieving control information that is used to execute a program;

a destination defining part defining destination address information ;

a moving part moving the program in accordance with the destination address information; and

a control information changing part changing the control information based on the destination address information.

2. The information processor as claimed in claim 1, wherein said control information comprises current address information indicating where the program is stored in a storage device, and

wherein said control information changing part comprises a replacing part replacing the current address information with the destination address information to which the program is moved.

3. The information processor as claimed in claim 1, wherein said control information is generated when said program is installed into a storage device.

4. The information processor as claimed in claim 1, wherein said control information is referred to when said program is executed.

5. The information processor as claimed in claim 1, wherein said control information is stored in a file referred to by other programs, and the file includes a plurality of control information to execute the other programs.

6. The information processor as claimed in claim 1, wherein said control information comprises definition information including at least one destination address information related to the program and including at least one definition name uniquely assigned to the destination address information, and

wherein said control information changing part comprises a changing part changing said control information based on said definition information.

7. The information processor as claimed in claim 1, wherein said control information comprises:

current address information indicating where the program is stored in a storage device;

definition information including at least one destination address information related to the program and including at least one definition name uniquely assigned to the destination address information,

wherein said moving part comprises:

a copying part retrieving the current address information corresponding to said definition name included in said control information in accordance with said definition information and copying all information, which is stored at a current address indicated by the current address information, in accordance with the destination address information; and

a deleting part deleting all information stored at the current address, and

wherein said control information changing part comprises:

a changing part changing the current address information included in said control information based on the destination address information included in said definition information.

8. The information processor as claimed in claim 1, wherein said control information comprises:

program information to execute the program; and

data information related to data created or edited by executing said program, and

wherein said moving part comprises:

a program moving part moving the program;

and

a data moving part moving the data when the program is moved by said program moving part.

9. The information processor as claimed in claim 1, further comprising an installing part installing said program.

10. A method for processing information comprising the steps of:

(a) retrieving control information that is used to execute a program;

(b) defining destination address information;

(c) moving the program in accordance with the destination address information; and

(d) changing the control information based on the destination address information.

11. The method as claimed in claim 10, wherein said control information includes current address information indicating where the program is stored in a storage device, and

wherein said step (d) comprises the step of replacing the current address information with the destination address information to which the program is moved.

12. The method as claimed in claim 10, wherein said control information is generated when said program is installed into a storage device.

13. The method as claimed in claim 10, wherein said control information is referred to when said program is executed.

14. The method as claimed in claim 10, wherein said control information is stored in a file referred to by other programs, and the file includes a plurality of control information to execute the other programs.

15. The method as claimed in claim 10, wherein said control information comprises definition information including at least one destination address information related to the program and including at least one definition name uniquely assigned to the destination address information, and

wherein said step (d) comprises the step of changing said control information based on said definition information.

16. The method as claimed in claim 10, wherein said control information comprises:

current address information indicating where the program is stored in a storage device;

definition information including at least one destination address information related to the program and including at least one definition name uniquely assigned to the destination address information,

wherein said step (c) comprises the steps of:

retrieving the current address information corresponding to said definition name included in said control information in accordance with said definition information

copying all information, which is stored at a current address indicated by the current address information, in accordance with the destination address information; and

deleting all information stored at the current address, and

wherein said step (d) comprises the steps of:

changing the current address information included in said control information based on the destination address information included in said definition information.

17. The method as claimed in claim 10, wherein said control information comprises program

information to execute the program and data information related to data created or edited by executing said program, and

wherein said step (c) comprises the steps of:

moving the program; and

moving the data when the program is moved.

18. The method as claimed in claim 10, further comprising the step of installing said program.

19. A computer-readable recording medium recorded with a program code for causing a computer to process information, said program code comprising the codes for:

(a) retrieving control information that is used to execute a program;

(b) defining destination address information;

(c) moving the program in accordance with the destination address information; and

(d) changing the control information based on the destination address information.

20. The computer-readable recording medium as claimed in claim 19, wherein said control information includes current address information

indicating where the program is stored in a storage device, and

wherein said code (d) comprises code for replacing the current address information with the destination address information to which the program is moved.

21. The computer-readable recording medium as claimed in claim 19, wherein said control information is generated when said program is installed into a storage device.

22. The computer-readable recording medium as claimed in claim 19, wherein said control information is referred to when said program is executed.

23. The computer-readable recording medium as claimed in claim 19, wherein said control information is stored in a file referred to by other programs and the file includes a plurality of control information to execute the other programs.

24. The computer-readable recording medium as claimed in claim 19, wherein said control

information comprises definition information including at least one destination address information related to the program and including at least one definition name uniquely assigned to the destination address information, and

wherein said code (d) comprises the code for changing said control information based on said definition information.

25. The computer-readable recording medium as claimed in claim 19, wherein said control information comprises:

current address information indicating where the program is stored in a storage device;

definition information including at least one destination address information related to the program and including at least one definition name uniquely assigned to the destination address information,

wherein said code (c) comprises the codes for:

retrieving the current address information corresponding to said definition name included in said control information in accordance with said definition information

copying all information, which is stored at a current address indicated by the current address information, in accordance with the destination address information; and

deleting all information stored at the current address, and

wherein said code (d) comprises the code for:

changing the current address information

included in said control information based on the destination address information included in said definition information.

26. The computer-readable recording medium as claimed in claim 19, wherein said control information comprises program information to execute the program and data information related to data created or edited by executing said program, and

wherein said code (c) comprises the codes for:

moving the program; and

moving the data when the program is moved.

27. The computer-readable recording medium as claimed in claim 19, further comprising the code for installing said program.

ABSTRACT OF THE DISCLOSURE

The present invention provides an information processor that can move an application easily and safely. In the information processor, a control information retrieving part retrieves the control information that is used to execute a program and a destination defining part defines destination address information to move the program. Then, a moving part moves the program in accordance with the destination address information. Also, a control information changing part changes the control information based on the destination address information.

FIG. 1 PRIOR ART

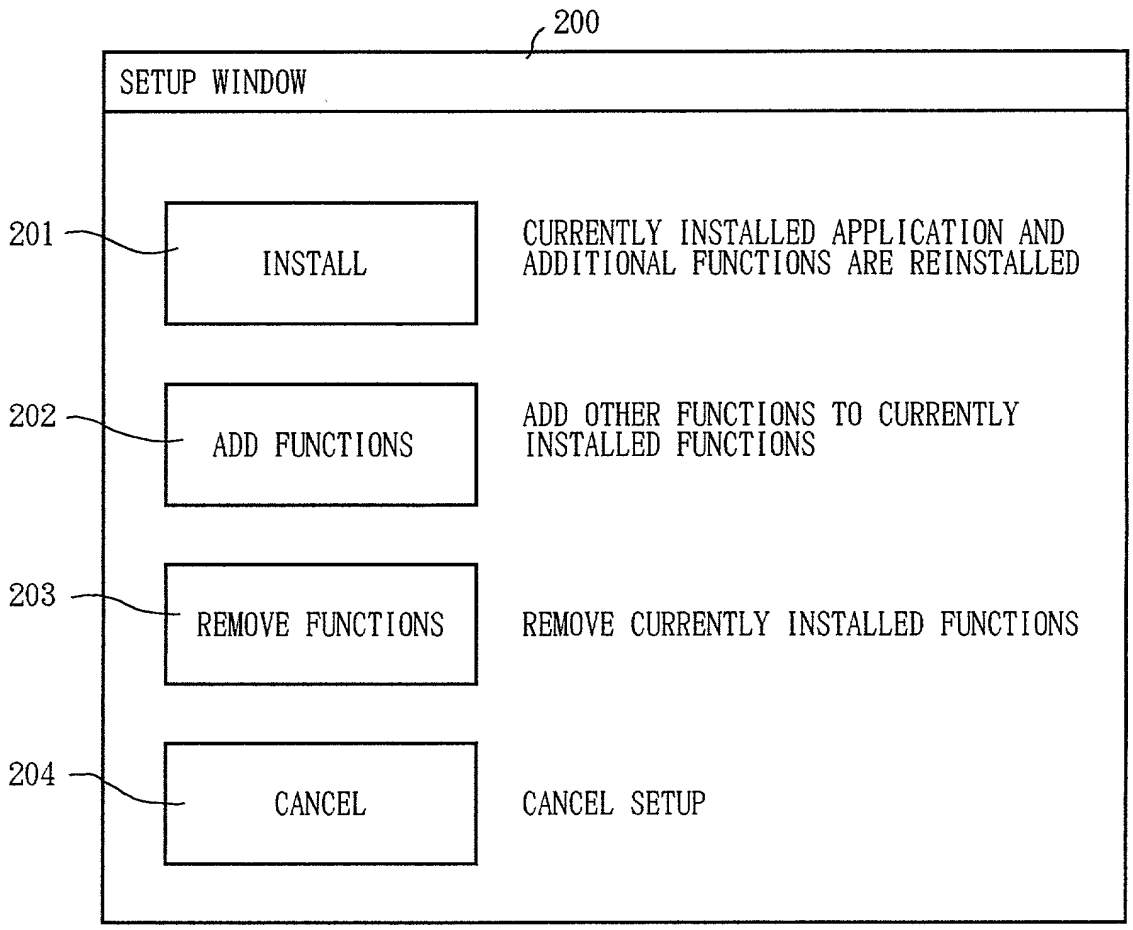


FIG. 2

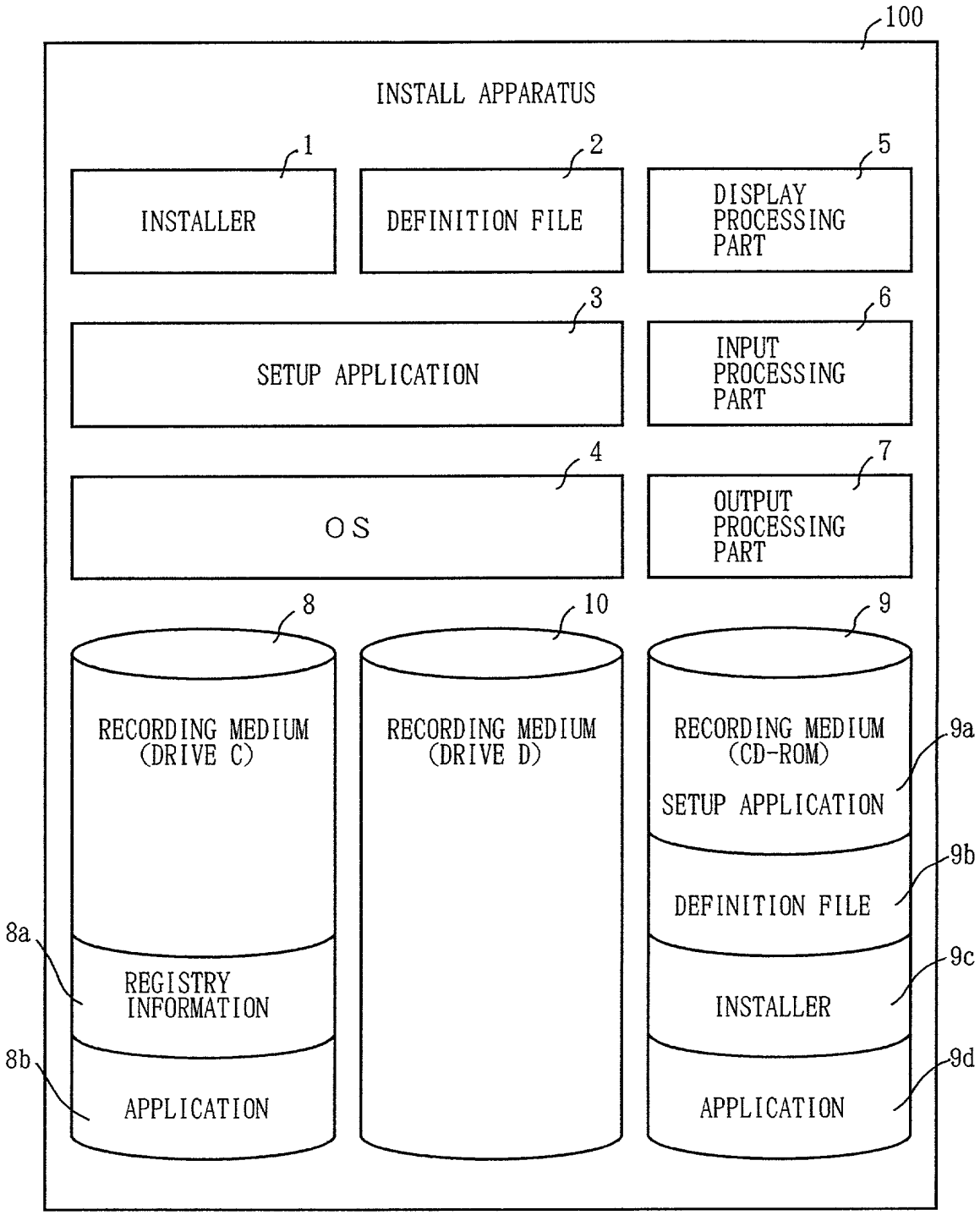

100
INSTALL APPARATUS
1
INSTALLER
2
DEFINITION FILE
5
DISPLAY PROCESSING PART
3
SETUP APPLICATION
6
INPUT PROCESSING PART
4
OS
7
OUTPUT PROCESSING PART
8
RECORDING MEDIUM (DRIVE C)
8a
REGISTRY INFORMATION
8b
APPLICATION
10
RECORDING MEDIUM (DRIVE D)
9
RECORDING MEDIUM (CD-ROM)
9a
SETUP APPLICATION
9b
DEFINITION FILE
9c
INSTALLER
9d
APPLICATION

FIG. 3

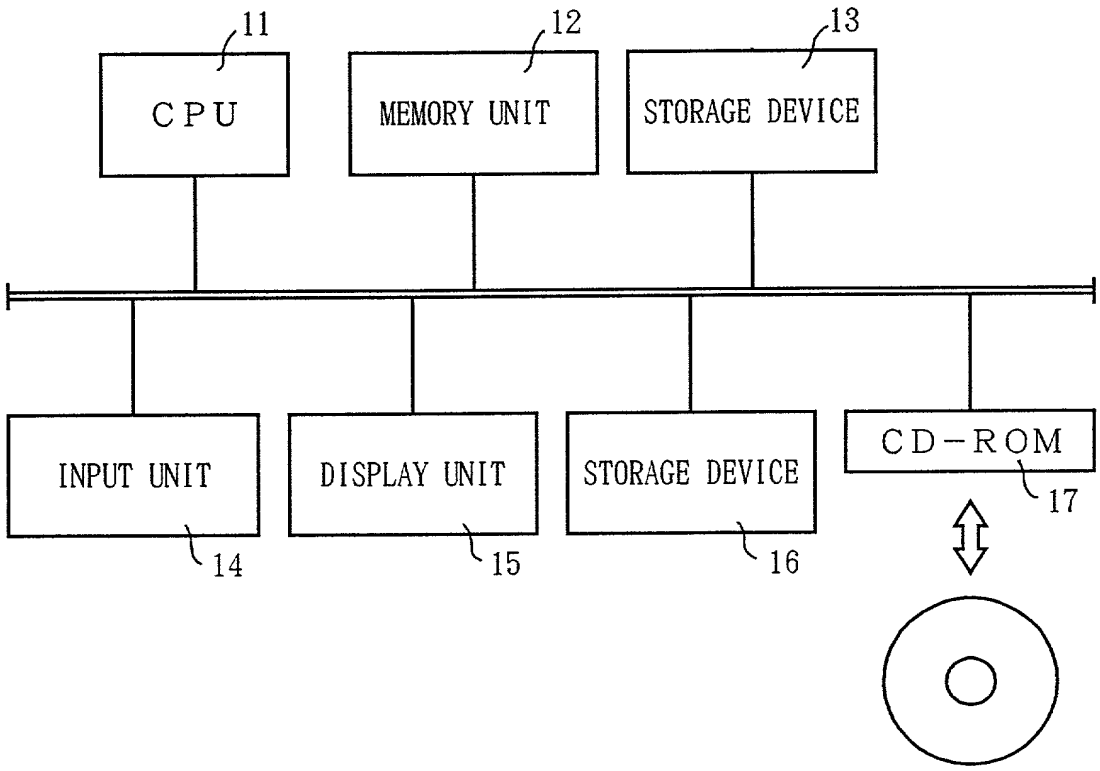

11
12
13
CPU
MEMORY UNIT
STORAGE DEVICE
INPUT UNIT
DISPLAY UNIT
STORAGE DEVICE
CD-ROM
14
15
16
17

FIG. 4A

BEFORE MOVE

REGISTRY INFORMATION 8 a

InstallDir	"C:¥ProgramFiles¥OmakaseV3"
DataPath	"C:¥ProgramFiles¥OmakaseV3¥Data"
ProgramFolder	"OmakaseV3"

FIG. 4B

AFTER MOVE

REGISTRY INFORMATION 8 a

InstallDir	"D:¥OmakaseV3"
DataPath	"D:¥OmakaseV3¥Data"
ProgramFolder	"OmakaseV3"

FIG. 5A

STRUCTURE OF DEFINITION FILE

```
[Path]
0=<ROOT KEY>,<SUB KEY>,<VALUE NAME>,<ADDITIONAL PATH>
1=<ROOT KEY>,<SUB KEY>,<VALUE NAME>,<ADDITIONAL PATH>

[Folder]
<ICON NAME>=<COMMAND LINE>,<FOLDER>,<ICON FILE>,<ICON INDEX>
```

FIG. 5B

EXAMPLE OF DEFINITINON FILE

```
[Path]
0=MACHINE-A,Software¥Fujitsu¥Omakase¥V3.0¥Dir,InstallDir,
1=MACHINE-A,Software¥Fujitsu¥Omakase¥V3.0¥Dir,DataPath,Data

[Folder]
OmakaseV3=%Install%¥Omakase.exe,%Install%,%Install%¥Omakase.exe,0
```

FIG. 6

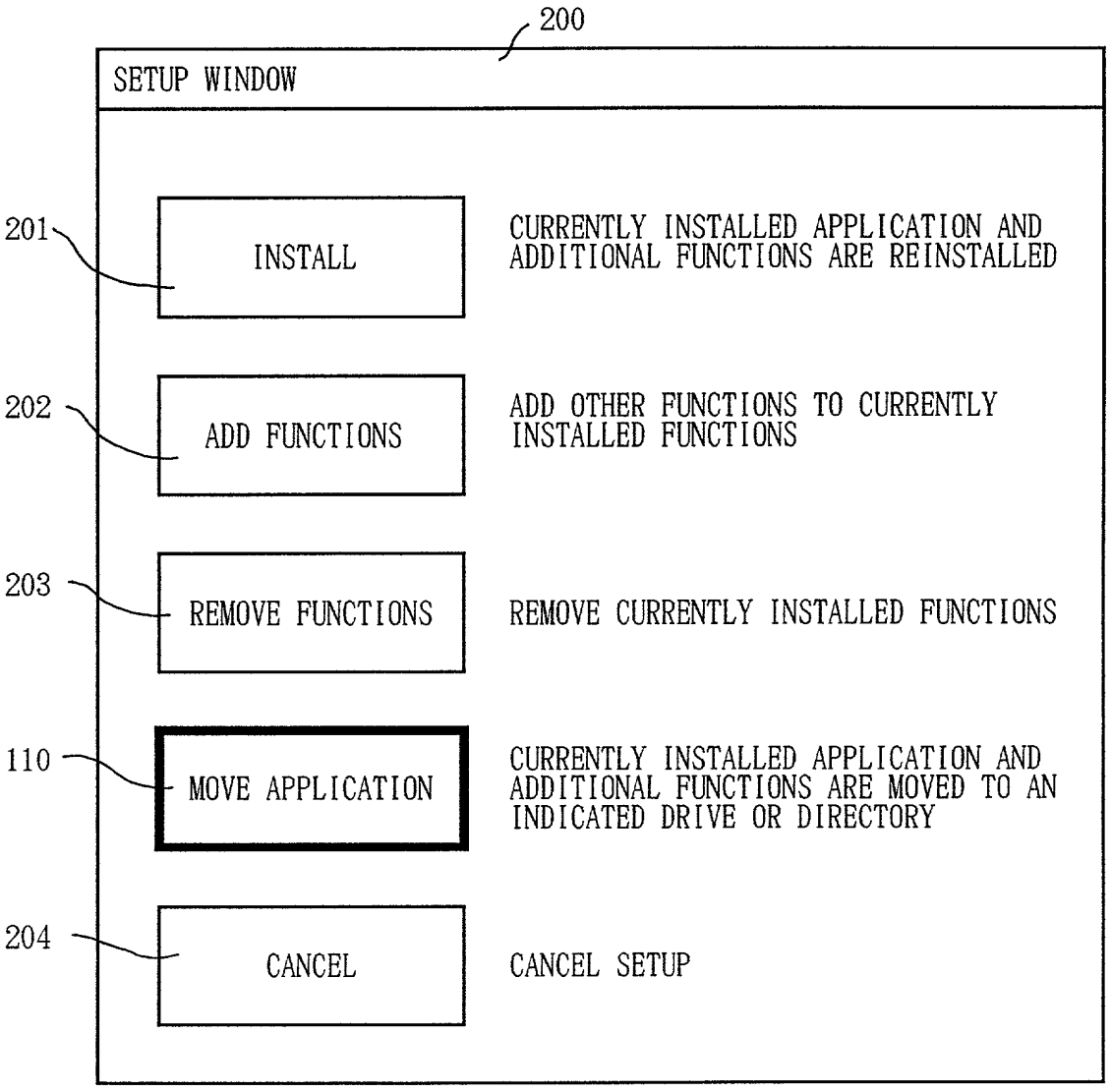

200
SETUP WINDOW
201
INSTALL
CURRENTLY INSTALLED APPLICATION AND ADDITIONAL FUNCTIONS ARE REINSTALLED
202
ADD FUNCTIONS
ADD OTHER FUNCTIONS TO CURRENTLY INSTALLED FUNCTIONS
203
REMOVE FUNCTIONS
REMOVE CURRENTLY INSTALLED FUNCTIONS
110
MOVE APPLICATION
CURRENTLY INSTALLED APPLICATION AND ADDITIONAL FUNCTIONS ARE MOVED TO AN INDICATED DRIVE OR DIRECTORY
204
CANCEL
CANCEL SETUP

FIG. 7

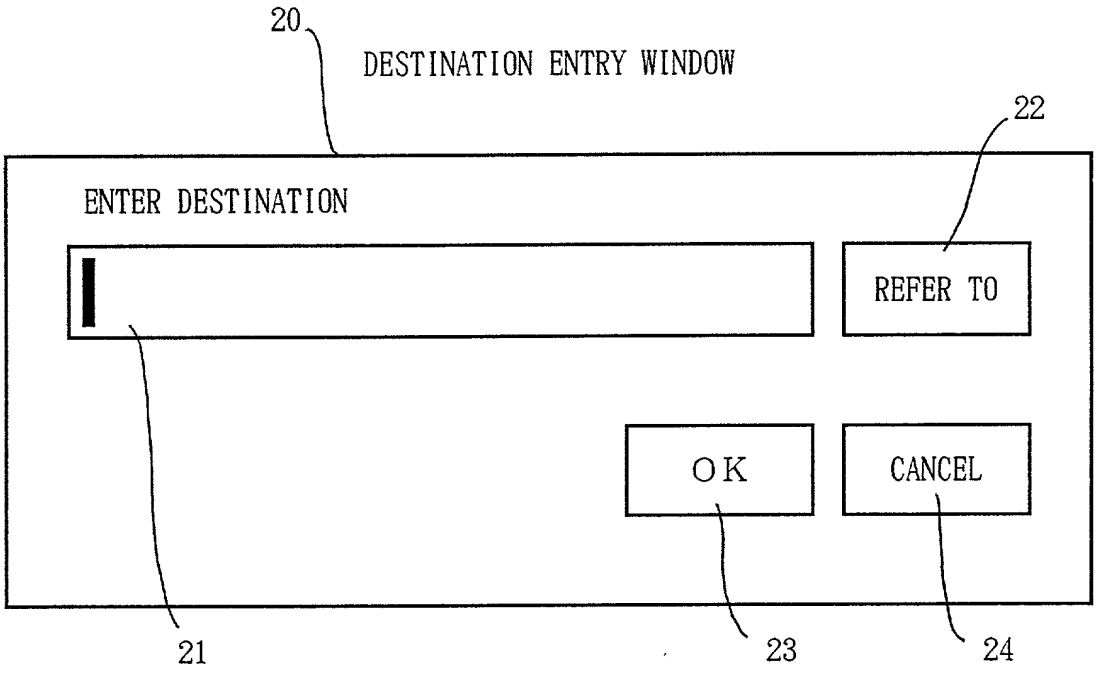

20
DESTINATION ENTRY WINDOW
22
ENTER DESTINATION
REFER TO
OK
CANCEL
21
23
24

FIG. 8

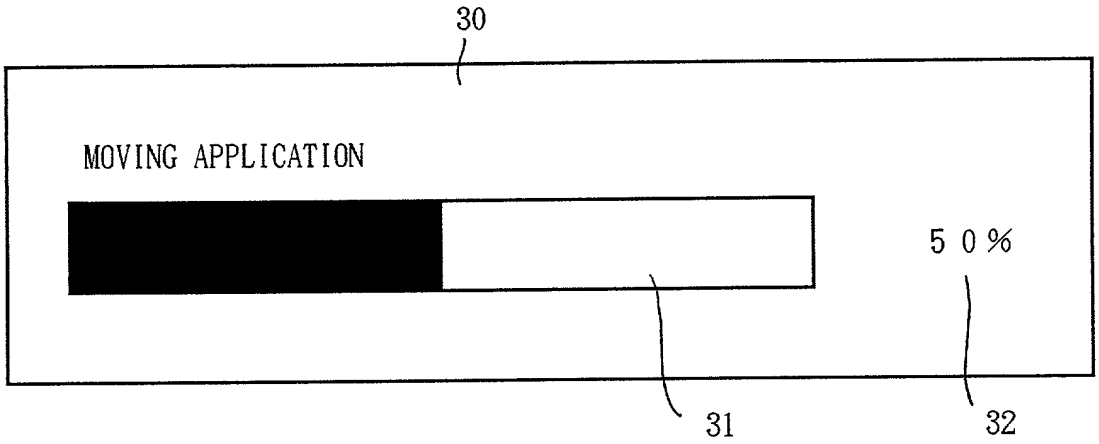
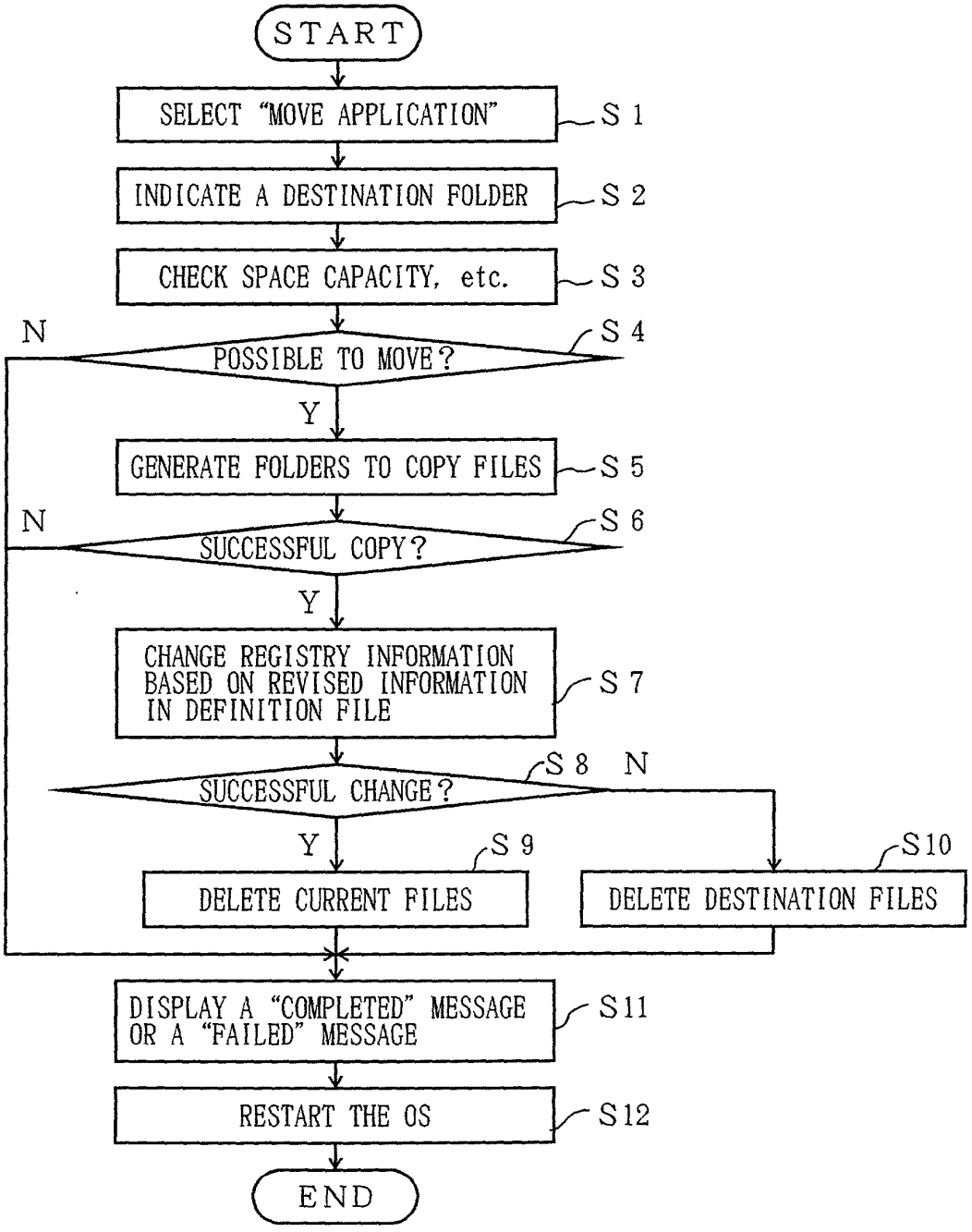


FIG. 9


START
SELECT "MOVE APPLICATION"
S 1
INDICATE A DESTINATION FOLDER
S 2
CHECK SPACE CAPACITY, etc.
S 3
N
POSSIBLE TO MOVE?
S 4
Y
GENERATE FOLDERS TO COPY FILES
S 5
N
SUCCESSFUL COPY?
S 6
Y
CHANGE REGISTRY INFORMATION BASED ON REVISED INFORMATION IN DEFINITION FILE
S 7
SUCCESSFUL CHANGE?
S 8
N
Y
S 9
DELETE CURRENT FILES
S10
DELETE DESTINATION FILES
DISPLAY A "COMPLETED" MESSAGE OR A "FAILED" MESSAGE
S11
RESTART THE OS
S12
END

PTO/SB/106 (8-96)
Approved for use through 9/30/98. OMB 0651-0032
Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE

Declaration and Power of Attorney For Patent Application

特許出願宣言書及び委任状

Japanese Language Declaration

日本語宣言書

下記の氏名の発明者として、私は以下の通り宣言します。

As a below named inventor, I hereby declare that:

私の住所、私書箱、国籍は下記の私の氏名の後に記載された通りです。

My residence, post office address and citizenship are as stated next to my name.

下記の名称の発明に関して請求範囲に記載され、特許出願している発明内容について、私が最初かつ唯一の発明者（下記の氏名が一つの場合）もしくは最初かつ共同発明者であると（下記の名称が複数の場合）信じています。

I believe I am the original, first and sole inventor (if only one name is listed below) or an original, first and joint inventor (if plural names are listed below) of the subject matter which is claimed and for which a patent is sought on the invention entitled

INFORMATION PROCESSOR, METHOD FOR PROCESSING INFORMATION AND COMPUTER-READABLE RECORDING MEDIUM RECORDED WITH PROGRAM CODE FOR CONTROLLING A COMPUTER TO PROCESS INFORMATION

上記発明の明細書（下記の欄でx印がついていない場合は、本書に添付）は、

the specification of which is attached hereto unless the following box is checked:

☐ __月__日に提出され、米国出願番号または特許協定条約国際出願番号を_____とし、
（該当する場合）_____ に訂正されました。

☐ was filed on _____
as United States Application Number or
PCT International Application Number
_____ and was amended on
_____ (if applicable).

私は、特許請求範囲を含む上記訂正後の明細書を検討し、内容を理解していることをここに表明します。

I hereby state that I have reviewed and understand the contents of the above identified specification, including the claims, as amended by any amendment referred to above.

私は、連邦規則法典第３７編第１条５６項に定義されるとおり、特許資格の有無について重要な情報を開示する義務があることを認めます。

I acknowledge the duty to disclose information which is material to patentability as defined in Title 37, Code of Federal Regulations, Section 1.56.

PTO/SB/106 (8-96)
Approved for use through 9/30/98. OMB 0651-0032
Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE

Japanese Language Declaration
(日本語宣言書)

私は、米国法典第３５編１１９条(a)-(d)項又は３６５条(b)項に基き下記の、米国以外の国の少なくとも一ヵ国を指定している特許協力条約３６５(a)項に基ずく国際出願、又は外国での特許出願もしくは発明者証の出願についての外国優先権をここに主張するとともに、優先権を主張している、本出願の前に出願された特許または発明者証の外国出願を以下に、枠内をマークすることで、示しています。

I hereby claim foreign priority under Title 35, United States Code, Section 119 (a)-(d) or 365(b) of any foreign application(s) for patent or inventor's certificate, or 365(a) of any PCT International application which designated at least one country other than the United States, listed below and have also identified below, by checking the box, any foreign application for patent or inventor's certificate, or PCT International application having a filing date before that of the application on which priority is claimed.

Prior Foreign Application(s)
外国での先行出願

(Number) (番号)	(Country) (国名)	(Day/Month/Year Filed) (出願年月日)	Priority Not Claimed 優先権主張なし
Pat. Appln. No.11-054179	Japan	2/March/1999	☐
			☐

私は、第３５編米国法典１１９条(e)項に基いて下記の米国特許出願規定に記載された権利をここに主張いたします。

I hereby claim the benefit under Title 35, United States Code, Section 119(e) of any United States provisional application(s) listed below.

(Application No.) (出願番号)	(Filing Date) (出願日)	(Application No.) (出願番号)	(Filing Date) (出願日)

私は、下記の米国法典第３５編１２０条に基いて下記の米国特許出願に記載された権利、又は米国を指定している特許協力条約３６５条(c)に基ずく権利をここに主張します。また、本出願の各請求範囲の内容が米国法典第３５編１１２条第１項又は特許協力条約で規定された方法で先行する米国特許出願に開示されていない限り、その先行米国出願書提出日以降で本出願書の日本国内または特許協力条約国際提出日までの期間中に入手された、連邦規則法典第３７編１条５６項で定義された特許資格の有無に関する重要な情報について開示義務があることを認識しています。

I hereby claim the benefit under Title 35, United States Code, Section 120 of any United States application(s), or 365(c) of any PCT International application designating the United States, listed below and, insofar as the subject matter of each of the claims of this application is not disclosed in the prior United States or PCT International application in the manner provided by the first paragraph of Title 35, United States Code Section 112, I acknowledge the duty to disclose information which is material to patentability as defined in Title 37, Code of Federal Regulations, Section 1.56 which became available between the filing date of the prior application and the national or PCT International filing date of application.

(Application No.) (出願番号)	(Filing Date) (出願日)	(Status: Patented, Pending, Abandoned) (現況：特許許可済、係属中、放棄済)

私は、私自身の知識に基づいて本宣言書中で私が行なう表明が真実であり、かつ私の入手した情報と私の信じるところに基ずく表明が全て真実であると信じていること、さらに故意になされた虚偽の表明及びそれと同等の行為は米国法典第１８編第１００１条に基ずき、罰金または拘禁、もしくはその両方により処罰されること、そしてそのような故意による虚偽の声明を行なえば、出願した、又は共に許可された特許の有効性が失われることを認識し、よってここに下記のごとく宣誓を致します。

I hereby declare that all statements made herein of my own knowledge are true and that all statements made on information and belief are believed to be true; and further that these statements were made with the knowledge that willful false statements and the like so made are punishable by fine or imprisonment, or both, under Section 1001 of Title 18 of the United States Code and that such willful false statements may jeopardize the validity of the application or any patent issued thereon.

PTO/SB/106 (8-96)
Approved for use through 9/30/98. OMB 0651-0032
Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE

Japanese Language Declaration
(日本語宣言書)

委任状： 私は下記の発明者として、本出願に関する一切の手続きを米特許商標局に対して遂行する弁理士または代理人として、下記の者を指名いたします。（弁護士、または代理人氏名及び登録番号を明記のこと）

POWER OF ATTORNEY: As a named inventor, I hereby appoint the following attorney(s) and/or agent(s) to prosecute this application and transact all business in the Patent and Trademark Office connected therewith *(list name and registration number)*

James D. Halsey, Jr., 22,729; Harry John Staas, 22,010; David M. Pitcher, 25,908; John C. Garvey, 28,607; J. Randall Beckers, 30,358; William F. Herbert, 31,024; Richard A. Gollhofer, 31,106; Mark J. Henry, 36,162; Gene M. Garner II, 34,172; Michael D. Stein, 37,240; Paul I. Kravetz, 35,230; Gerald P. Joyce, III, 37,648; Todd E. Marlette, 35,269; Harlan B. Williams, Jr., 34,756; George N. Stevens, 36,938; Michael C. Soldner, P-41,455 and William M. Schertler, 35,348 (agent)

書類送付先

Send Correspondence to:

STAAS & HALSEY
700 Eleventh Street, N.W.
Suite 500
Washington, D.C. 20001

直接電話連絡先： （名前及び電話番号）

Direct Telephone Calls to: *(name and telephone number)*

STAAS & HALSEY
(202) 434-1500

唯一または第一発明者名		Full name of sole or first inventor	Yuji Kumakura
発明者の署名	日付	Inventor's signature	Yuji Kumakura Date January 13, 2000
住所		Residence	Nagaoka-shi, Niigata, Japan
国籍		Citizenship	Japan
私書箱		Post Office Address	c/o FUJITSU OASYS DEVELOPMENT CO., LTD,
			1-1, Higashi-Sakanoue 2-chome, Nagaoka-shi, Niigata, 940-0066 Japan
第二共同発明者		Full name of second joint inventor, if any	
第二共同発明者	日付	Second inventor's signature	Date
住所		Residence	
国籍		Citizenship	
私書箱		Post Office Address	

（第三以降の共同発明者についても同様に記載し、署名をすること）

(Supply similar information and signature for third and subsequent joint inventors.)